(19) Europäisches Patentamt
European Patent Office
Office européen des brevets

(11) EP 3 099 003 B1

(12) **EUROPEAN PATENT SPECIFICATION**

(45) Date of publication and mention of the grant of the patent:
27.01.2021 Bulletin 2021/04

(51) Int Cl.:
H04L 9/08 *(2006.01)*

(21) Application number: **16169448.4**

(22) Date of filing: **12.05.2016**

(54) **EFFICIENT KEY DERIVATION WITH FORWARD SECRECY**

EFFIZIENTE SCHLÜSSELABLEITUNG MIT VORWÄRTSSICHERHEIT

DÉRIVATION DE CLÉ EFFICACE AVEC CONFIDENTIALITÉ PERSISTANTE

(84) Designated Contracting States:
AL AT BE BG CH CY CZ DE DK EE ES FI FR GB GR HR HU IE IS IT LI LT LU LV MC MK MT NL NO PL PT RO RS SE SI SK SM TR

(30) Priority: **28.05.2015 US 201514724315**

(43) Date of publication of application:
30.11.2016 Bulletin 2016/48

(73) Proprietor: **NXP B.V.**
5656 AG Eindhoven (NL)

(72) Inventors:
- **PEETERS, Michaël**
  5656 AG Eindhoven (NL)
- **VERSLEGERS, Rudi**
  5656 AG Eindhoven (NL)
- **WARNEZ, Dimitri**
  5656 AG Eindhoven (NL)

(74) Representative: **Krott, Michel**
NXP Semiconductors
Intellectual Property & Licensing
High Tech Campus 60
5656 AG Eindhoven (NL)

(56) References cited:
US-A1- 2007 003 063

- **Elaine Barker ET AL: "NIST Special Publication 800-57: Recommendation for Key Management Part 1: General, revision 3", Reommendation for Key Management Part 1: General, 1 July 2012 (2012-07-01), XP055307975, DOI: 10.6028/NIST.SP.800-57pt1r3 Retrieved from the Internet: URL:http://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-57p1r3.pdf [retrieved on 2016-10-05]**
- **"Chapter 13: Key Management Techniques ED - Menezes A J; Van Oorschot P C; Vanstone S A", HANDBOOK OF APPLIED CRYPTOGRAPHY; [CRC PRESS SERIES ON DISCRETE MATHEMATICES AND ITS APPLICATIONS], CRC PRESS, BOCA RATON, FL, US, PAGE(S) 543 - 590 , 1 October 1996 (1996-10-01), XP001525013, ISBN: 978-0-8493-8523-0 Retrieved from the Internet: URL:http://www.cacr.math.uwaterloo.ca/hac/**

EP 3 099 003 B1

Description

TECHNICAL FIELD

[0001] Various embodiments disclosed herein relate generally to secure channel communications and, more particularly but not exclusively, to session key derivation and master key secrecy.

BACKGROUND

[0002] When two devices want to communicate securely, they typically first establish a secure channel. This channel will provide protection for confidentiality, integrity, and/or authentication of the exchanged messages. The method of choice to implement this secure channel is usually based on symmetric cryptography. In that setting, both devices first share a common and secret symmetric key usually called the master key. The master key is typically either predefined at manufacturing time or obtained via public key-based key exchange protocols (for instance elliptic curve Diffie-Hellman). In well-designed secure channels, the master key is never used as-is to encrypt, authenticate, or sign the messages. Instead, the master key is used to generate so-called session keys that are then each dedicated to a particular task (like encryption or authentication) for a limited period of time.

[0003] US 2007/0003063 A1 describes methods and apparatus to perform associated extensions for negotiated channel security protocols. A described method to extend a security protocol comprises exchanging identifying information between a first and a second endpoint, determining a secret based on the exchanged identifying information, determining a first master secret based on the determined secret and a second master secret determined in a prior protocol exchange block, and deriving a session key based on the first master secret.

SUMMARY

[0004] A brief summary of various embodiments is presented below. Some simplifications and omissions may be made in the following summary, which is intended to highlight and introduce some aspects of the various embodiments, but not to limit the scope of the invention. Detailed descriptions of a preferred embodiment adequate to allow those of ordinary skill in the art to make and use the inventive concepts will follow in later sections.

[0005] Various embodiments described herein relate a method for execution by a processor of a hardware system for performing secure communication via a secure communication channel as defined in claim 1, a non-transitory machine-readable medium comprising instructions as defined in claim 6, and a system for performing secure communication via a secure communication channel as defined in claim 8. Further embodiments are defined in the dependent claims.

BRIEF DESCRIPTION OF THE DRAWINGS

[0006] In order to better understand various embodiments, reference is made to the accompanying drawings, wherein:

FIG. 1 illustrates an example of a hardware system for performing secure communications;
FIG. 2 illustrates a first example of a method for generating session keys and a new master key;
FIG. 3 illustrates a second example of a method for generating session keys and a new master key;
FIG. 4 illustrates a third example of a method for generating a session key and a new master key; and
FIG. 5 illustrates a fourth example of a method for generating a session key and a new master key.

[0007] To facilitate understanding, identical reference numerals have been used to designate elements having substantially the same or similar structure or substantially the same or similar function.

DETAILED DESCRIPTION

[0008] The description and drawings presented herein illustrate various principles. It will be appreciated that those skilled in the art will be able to devise various arrangements that, although not explicitly described or shown herein, embody these principles and are included within the scope of this disclosure. As used herein, the term, "or" refers to a non-exclusive or (*i.e.*, and/or), unless otherwise indicated (*e.g.*, "or else" or "or in the alternative"). Additionally, the various embodiments described herein are not necessarily mutually exclusive and may be combined to produce additional embodiments that incorporate the principles described herein.

[0009] As noted above, many secure communications implementations utilize a shared master key between devices which is then used to generate session keys that are actually used for tasks such as encryption and digital signing. If the master key is compromised, however, an attacker will be able to generate the session keys themselves and access the content of any intercepted communications, including both future and past communications.

[0010] Various embodiments described herein attempt to provide forward secrecy for secured communications by periodically altering the master key and discarding all old versions of the master key. Both parties to a communication alter the master key in the same way, such that renegotiation of the master key need not be performed; instead, both devices modify the master key according to the same protocol and therefore always hold the same version of the master key. In this way, even if the current master key is compromised, the attacker does not have access to previous versions of the master key and therefore may not be able to decrypt previously-intercepted communications.

[0011] FIG. 1 illustrates an example of a hardware system 100 for performing secure communications. The hardware system 100 may be virtually any device that communicates using an encrypted protocol. As shown, the device 100 includes a processor 120, memory 130, secure element 140, network interface 150, and storage 160 interconnected via one or more system buses 110. It will be understood that FIG. 1 constitutes, in some respects, an abstraction and that the actual organization of the components of the device 100 may be more complex than illustrated.

[0012] The processor 120 may be any hardware device capable of executing instructions stored in memory 130 or storage 160 or otherwise processing data. As such, the processor may include a microprocessor, field programmable gate array (FPGA), application-specific integrated circuit (ASIC), or other similar devices.

[0013] The memory 130 may include various memories such as, for example LI, L2, or L3 cache or system memory. As such, the memory 130 may include static random access memory (SRAM), dynamic RAM (DRAM), flash memory, read only memory (ROM), or other similar memory devices.

[0014] The secure element 140 may be a device for performing secured functions such as, for example, storing secret information or performing cryptographic processing. As such, the secure element 140 may include hardware such as a processor, a memory, and an interface to the bus 110. The rest of the hardware device 100 may be referred to as a host device to the secure element. In various embodiments, the secure element 140 and processor 120 both implement a session key derivation method, such as one of the examples described herein, to enable secure communications therebetween. Alternatively, only one of the secure element 140 and processor 120 may implement such a method along with another external device which is accessible via the network interface.

[0015] The network interface 150 may include one or more devices for enabling communication with other hardware devices. For example, the network interface 150 may include a network interface card (NIC) configured to communicate according to the Ethernet protocol. Additionally, the network interface 150 may implement a TCP/IP stack for communication according to the TCP/IP protocols. Additionally or alternatively, the network interface may include a near-field communication (NFC) chip, 3G chip, or 4G chip along with appropriate antennas to enable wireless communications. Various alternative or additional hardware or configurations for the network interface 150 will be apparent.

[0016] The storage 160 may include one or more machine-readable storage media such as read-only memory (ROM), random-access memory (RAM), magnetic disk storage media, optical storage media, flash-memory devices, or similar storage media. In various embodiments, the storage 160 may store instructions for execution by the processor 120 or data upon with the processor 120 may operate. For example, the storage 160 may store a base operating system 161 for controlling various basic operations of the hardware 100 and multiple applications 162 for performing various functions on behalf of the user. For example, the applications 162 may include a web browser, a music player, or a wireless payment application (or a non-secured portion thereof, with the secured portion residing on the secure element 140).

[0017] To enable secure communications (*e.g.* with the secure element 140 or via the network interface 150), the storage 160 also stores a master key 163 and multiple session keys 164. In various embodiments, the initial value of the master key 163 is pre-configured as stored in the storage 160 at the time of device manufacturing while in other embodiments the initial value of the master key 163 is negotiated between two devices according to an appropriate key exchange protocol. In the latter embodiments, the storage 160 includes master key exchange instructions 165 that transmit a master key value to or receive a master key value from another device. The storage 160 also includes encryption instructions 166 such as, for example, an AES implementation that can be used in conjunction with the session keys 166 to encrypt communications.

[0018] Periodically, new session keys 164 are generated (at both ends of the secure communication channel) to reduce the likelihood that the session keys 164 will be compromised. As such, the storage 160 includes session key generation instructions 167. To provide forward secrecy, the session key generation instructions 167 also include master key modification instructions 168 that periodically alter the master key 163 value. For example, the master key modification instructions 168 may generate a new master key value and subsequently delete the current master key value by, for example, overwriting the current master key value with the new master key value or other data or by freeing an area in memory storing the current master key value. Various example embodiments of the session key generation instructions 167 and master key modification instructions 168 will be described below.

[0019] It will be apparent that various information described as stored in the storage 160 may be additionally or alternatively stored in the memory 130. In this respect, the memory 130 may also be considered to constitute a "storage device" and the storage 160 may be considered a "memory." Various other arrangements will be apparent. Further, the memory 130 and storage 160 may both be considered to be "non-transitory machine-readable media." As used herein, the term "non-transitory" will be understood to exclude transitory signals but to include all forms of storage, including both volatile and non-volatile memories.

[0020] While the host device 100 is shown as including one of each described component, the various components may be duplicated in various embodiments. For example, the processor 120 may include multiple microprocessors that are configured to independently execute the methods described herein or are configured to perform steps or subroutines of the methods described herein such that the multiple processors cooperate to achieve the functionality described herein. Further, where the device 100 is implemented in a cloud computing system, the various hardware components may belong to separate physical systems. For example, the processor 120 may include a first processor in a first server and a second processor in a second server.

[0021] FIG. 2 illustrates a first example of a method 200 for generating session keys and a new master key. The method 200 may correspond to the session key generation instructions 167 of FIG. 1. As shown, the method 200 utilizes a current master key 210 to generate two session keys 240, 242. Specifically, the method 200 encrypts two constants 220, 222 using an AES encryption algorithm (*e.g.,* in ECB mode applied to one block) 230, 232 and the current master key 210. The constants 220, 222 may be any values that two implementations of the method 200 agree will be used for generation of the session keys 240, 242. For example, these constants 220, 222 may be pre-configured literal values, agreed-upon variables, or a combination thereof. For example, "CST_A" 220 may be a counter value appended to a device identifier, while "CST_B" 222 may be a current time (*e.g.*, the current hour and minute) appended to the device identifier. It will be apparent that virtually any value may be used as a constant 220, 222 as long as both parties to the communication are agreed as to which values will be used. In various embodiments, the selected constants need not be kept secret.

[0022] After the session keys 240, 242 have been generated, they may be used as part of ordinary cryptographic operations 260 performed by the operating system or applications. For example, as shown, the session keys 240, 242 may be used by other AES processes 264, 274 to encrypt plaintext 262, 272 for transmission to another device as ciphertext 266, 276 (or to decrypt ciphertext 266, 276 into usable plaintext 262, 274). It will be understood that additional session keys (not shown) may be generated in similar, agreed-upon manners.

[0023] Also after session key generation, the method 200 generates a new master key 212 to replace the current master key 210 by combining the two session keys 240, 242. Specifically, as shown, the session keys are combined using an exclusive or (XOR) operation 250. It will be appreciated that other combination functions may be used such as a not-and (NAND) operation, encryption of one session key using another, or any other transformative function. Additionally, it will be appreciated that more than two session keys may be combined to form the new master key value 212. For example, four session keys may be XOR'ed together to produce a single master key value.

[0024] After the new master key value 212 is generated, it replaces the old master key value 210. For example, the method may overwrite a master key variable or space in memory with the new value 212, thereby deleting the old master key value 210. By deleting the old master key value 210, the chances of the old master key value being compromised (and, consequently, the chances of the session keys 240, 242 being compromised) are reduced. As such, the method 200 provides forward secrecy. It will be appreciated that the method 200 may be executed periodically to refresh the session keys 240, 242 and, as such, the value of the master key may periodically change.

[0025] With knowledge of the method 200 used, an attacker that has compromised the session keys 240, 242 may generate the next master key 212 and, as such, may be able to generate each new session key in the future. To help combat this, various implementations may also utilize the old value of the master key in generating the new master key; as such, the attacker would also have to compromise the old value of the master key, which is generally protected using different, and often more stringent, techniques from those used to protect the session key.

[0026] FIG. 3 illustrates a second example of a method 300 for generating session keys and a new master key. The method 300 may correspond to the session key generation instructions 167 of FIG. 1. As can be seen, the method 300 is largely similar to the previous example method 200. A master key 310 is used to encrypt 330, 332 agreed-upon constants 320, 322 to produce session keys 340, 342. The session keys 340, 342 may then be used for secure communication operations 360 by the OS or applications to, for example, encrypt 364, 374 plaintext communications 362, 372 to produce encrypted communications 366, 376 (or to perform the reverse process of decryption).

[0027] To produce a new master key value 312, the method 300 combines the old master key value 310 with one or more of the session keys 340, 342. As shown, the master key 310 is combined with a first session key 340 via an XOR operation 350, the result of which is then combined with a second session key 342 via another XOR operation 352. As before, fewer or additional session keys may utilized and alternative or additional combination techniques may be used.

[0028] FIG. 4 illustrates a third example of a method 400 for generating a session key and a new master key. The method 400 may correspond to the session key generation instructions 167 of FIG. 1. As shown, the method is similar to the previous example method 300. The method 400 uses a master key 410 to encrypt 430 an agreed-upon constant 420 to produce a session key 440. The session key 440 may then be used for secure communication operations 460 by the OS or applications to, for example, encrypt 464 plaintext communications 462 to produce encrypted communications 466 (or to perform the reverse process of decryption).

[0029] To produce a new master key value 412, the method 400 combines the old master key value 410 with one session key 440. As shown, the combination operation is an XOR 450. As before, fewer or additional session keys may utilized and alternative or additional combination techniques may be used.

[0030] Thus, the third example method 400 is similar to the second example method 300 with the difference that the new master key 412 may be generated using only a single session key (and the old value for the master key). As such, the method 400 may be used to modify the master key after generation of each new session key. Alternatively, the method 400 may only be used to modify the master key when a session key for a specific function is generated. For example, if an implementation generates different session keys for encryption and authentication, the method 400 may only be used to change the master key whenever the session key for the encryption task (and not for the authentication task) is generated. Various other modifications will be apparent.

[0031] FIG. 5 illustrates a fourth example of a method 500 for generating a session key and a new master key. The method 500 may correspond to the session key generation instructions 167 of FIG. 1. This fourth example method 500 illustrates the use of a different combination function to create the new master key (as well as an alternative method to create a session key).

[0032] As shown, the method 500 first uses the current master key value 510 to generate a new master key value 512. Specifically, the method 500 encrypts 530 a constant value 520 that, as with previously-described constant values 220, 222, 320, 322, 420 is agreed-upon by the parties to the secure communication and between which the master key is shared. The result of the encryption step 530 is taken as the new value of the master key 512.

[0033] After the new master key 512 is generated, it can be used along with the old master key 510 to generate a new session key 540. Specifically, as shown, the two master keys 510, 512 are combined via an XOR operation 550 to produce a session key 540. The session key 540 may then be used for secure communication operations 560 by the OS or applications to, for example, encrypt 564 plaintext communications 562 to produce encrypted communications 566 (or to perform the reverse process of decryption). After the session key 540 is generated, the method 500 may delete the old master key value 510 to further guard against the value being compromised.

[0034] The method 500 may be used to modify the master key with the generation of each new session key. Alternatively, the method 500 may only be used to modify the master key when a session key for a specific function is to be generated. For example, if an implementation generates different session keys for encryption and authentication, the method 500 may only be used to change the master key whenever the session key for the encryption task (and not for the authentication task) is generated. As yet another alternative, the method 500 may be modified to not produce any session key at all, executing only the encryption operation 530 to generate the new master key 512. For example, the parties to the communication may be configured to periodically alter their master key based on such a modified method. Various other modifications will be apparent.

[0035] According to the foregoing, various embodiments enable the efficient derivation of session keys with the benefit of forward secrecy. For example, by periodically generating a new master key according to an agreed-upon process, the parties to a communication can provide forward secrecy to the master key without communicating each new master key to each other. Further, by using agreed-upon constants and simple combination operations to generate master and session keys, various embodiments are able to quickly and efficiently derive encryption keys necessary for secure communications. Such embodiments may be particularly beneficial in environments where secure communications are to be performed very quickly such as, for example, an "instant on" scenario (*e.g.*, a mobile device is activated or unlocked by a fingerprint scan which is compared against a fingerprint key stored in a secure element of the mobile device). Various additional benefits will be apparent in view of the foregoing.

[0036] It should be apparent from the foregoing description that various embodiments of the invention may be implemented in hardware. Furthermore, various embodiments may be implemented as instructions stored on a non-transitory machine-readable storage medium, such as a volatile or non-volatile memory, which may be read and executed by at least one processor to perform the operations described in detail herein. A machine-readable storage medium may include any mechanism for storing information in a form readable by a machine, such as a personal or laptop computer, a server, or other computing device. Thus, a non-transitory machine-readable storage medium excludes transitory signals but may include both volatile and non-volatile memories, including but not limited to read-only memory (ROM), random-access memory (RAM), magnetic disk storage media, optical storage media, flash-memory devices, and similar storage media.

[0037] It should be appreciated by those skilled in the art that any block diagrams herein represent conceptual views of illustrative circuitry embodying the principles of the invention. Similarly, it will be appreciated that any flow charts, flow diagrams, state transition diagrams, pseudo code, and the like represent various processes which may be substantially represented in machine readable media and so executed by a computer or processor, whether or not such computer or processor is explicitly shown.

[0038] Although the various embodiments have been described in detail with particular reference to certain aspects thereof, it should be understood that the invention is capable of other embodiments and its details are capable of modifications in various obvious respects. As is readily apparent to those skilled in the art, variations and modifications can be effected while remaining within the scope of the invention as defined by the claims. Accordingly, the foregoing disclosure, description, and figures are for illustrative purposes only and do not in any way limit the invention, which is defined only by the claims.

Claims

1. A method (200, 300, 500) for execution by a processor (120) of a hardware system (100) for performing secure communication via a secure communication channel with an external device, the method comprising:

    determining a master key value (210, 310, 510) for use in secure communications (260, 360, 560) with the external device;
    both in the hardware system (100) and in the external device, periodically deriving a first session key (240, 340, 540) using the master key value (210, 310, 510), and periodically generating a new master key value (212, 312, 512) based on the master key value (210, 310, 510); and
    deleting the master key value (210, 310, 510),
    wherein the periodically deriving the first session key (240, 340, 540) using the master key value (210, 310, 510) and the periodically generating the new master key value (212, 312, 512) based on the master key value (210, 310, 510) comprises one of the following procedures a), b), and c):

    a) deriving (232) a second session key (242) using the master key value (210), and combining (250) the derived first session key (240) and the derived second session key (242) to thereby generate the new master key value (212);
    b) deriving (332) a second session key (342) using the master key value (310), combining the master key value (310) and the derived first session key (340) to yield a combined result, and combining (352) the combined result and the derived second session key (342) to thereby generate the new master key value (312); and
    c) generating (530) the new master key value (512) using the master key value (510), and deriving the first or a second session key (540) by combining (550) the master key value (510) and the new master key value (512),

    and wherein the method further comprises both parties to the secure communication, that is the processor (120) and the external device, altering the master key in the same way.

2. The method (500) of procedure c) of claim 1, wherein generating a new master key value (512) comprises encrypting a constant value (520) using the master key value (510).

3. The method (200) of claim 1, wherein, in procedure a), the generating the new master key value (212) comprises applying an XOR function to the first derived session key and at least the second derived session key (242) derived using the master key value (210).

4. The method (200, 300) of any preceding claim, wherein deriving a session key (240, 242; 340, 342) using the master key value (210, 310) comprises:
encrypting a constant value (220, 222; 320, 322) using the master key value (210, 310).

5. The method (200, 300, 500) of any preceding claim, wherein determining a master key value (210, 310, 510) for use in secure communications (360, 560) with an external device comprises:
negotiating a master key value (210, 310, 510) with the external device, which comprises negotiating with a secure element (140) that is part of the same device as the processor (120).

6. A non-transitory machine-readable medium (160) comprising instructions (168) which, when executed on two computers, cause the two computers to perform the method of any preceding claim.

7. The non-transitory machine-readable medium (160) of claim 6, wherein the non-transitory machine-readable medium (160) is encoded with instructions for execution by a secure element (140) within a host device (100), being one of the two computers.

8. A system for performing secure communication via a secure communication channel between a hardware system (100) and an external device, the hardware system (100) including a processor (120), the processor (120) and the external device being configured
to determine a master key value (210, 310, 510) for use in secure communications (260, 360, 560) with the external device,
both in the hardware system (100) and in the external device, to periodically derive a first session key (240, 340, 540) using the master key value (210, 310, 510), and to periodically generate a new master key value (212, 312, 512) based on the master key value (210, 310, 510), and
to delete the current master key value (210, 310, 510),
wherein the processor (120) and the external device are further configured to periodically derive the first session key (240, 340, 540) using the master key

value (210, 310, 510) and to periodically generate the new master key value (212, 312, 512) based on the master key value (210, 310, 510) according to one of the following procedures a), b), and c):

a) deriving (232) a second session key (242) using the master key value (210), and combining (250) the derived first session key (240) and the derived second session key (242) to thereby generate the new master key value (212);
b) deriving (332) a second session key (342) using the master key value (310), combining the master key value (310) and the derived first session key (340) to yield a combined result, and combining (352) the combined result and the derived second session key (342) to thereby generate the new master key value (312); and
c) generating (530) the new master key value (512) using the master key value (510), and deriving the first or a second session key (540) by combining (550) the master key value (510) and the new master key value (512),

and wherein both parties to the secure communication, that is the processor (120) and the external device, are configured to alter the master key in the same way.

9. The system of procedure c) of claim 8, wherein in generating a new master key value (212, 312, 412, 512), the processor (120) is configured to encrypt a constant value (220, 320, 420, 520) using the master key value (210, 310, 410, 510).

10. The system of claim 8, wherein, in procedure a), the generating the new master key value (212) comprises applying an XOR function to the first derived session key and at least the second derived session key (242) derived using the master key value (210).

11. The system of claim 8, wherein in deriving a session key (240, 242; 340, 342) using the master key value (210, 310), the processor (120) is configured to encrypt a constant value (220, 222; 320, 322) using the master key value (210, 310).

Patentansprüche

1. Verfahren (200, 300, 500) zur Ausführung durch einen Prozessor (120) eines Hardwaresystems (100) zum Durchführen sicherer Kommunikation über einen sicheren Kommunikationskanal mit einer externen Vorrichtung, wobei das Verfahren umfasst:

Bestimmen eines Masterschlüsselwertes (210, 310, 510) zur Verwendung in sicheren Kommunikationen (260, 360, 560) mit der externen Vorrichtung;
sowohl in dem Hardwaresystem (100) als auch in der externen Vorrichtung periodisches Ableiten eines ersten Sitzungsschlüssels (240, 340, 540) unter Verwendung des Masterschlüsselwertes (210, 310, 510) und periodisches Erzeugen eines neuen Masterschlüsselwertes (212, 312, 512) auf Grundlage des Masterschlüsselwertes (210, 310, 510); und
Löschen des Masterschlüsselwertes (210, 310, 510),
wobei das periodische Ableiten des ersten Sitzungsschlüssels (240, 340, 540) unter Verwendung des Masterschlüsselwertes (210, 310, 510) und das periodische Erzeugen des neuen Masterschlüsselwertes (212, 312, 512) auf Grundlage des Masterschlüsselwertes (210, 310, 510) eine der folgenden Prozeduren a), b) und c) umfasst:

a) Ableiten (232) eines zweiten Sitzungsschlüssels (242) unter Verwendung des Masterschlüsselwertes (210) und Kombinieren (250) des abgeleiteten ersten Sitzungsschlüssels (240) und des abgeleiteten zweiten Sitzungsschlüssels (242), um dadurch den neuen Masterschlüsselwert (212) zu erzeugen;
b) Ableiten (332) eines zweiten Sitzungsschlüssels (342) unter Verwendung des Masterschlüsselwertes (310), Kombinieren des Masterschlüsselwertes (310) und des abgeleiteten ersten Sitzungsschlüssels (340), um ein kombiniertes Ergebnis zu erbringen, und Kombinieren (352) des kombinierten Ergebnisses und des abgeleiteten zweiten Sitzungsschlüssels (342), um dadurch den neuen Masterschlüsselwert (312) zu erzeugen; und
c) Erzeugen (530) des neuen Masterschlüsselwertes (512) unter Verwendung des Masterschlüsselwertes (510) und Ableiten des ersten oder eines zweiten Sitzungsschlüssels (540) durch Kombinieren (550) des Masterschlüsselwertes (510) und des neuen Masterschlüsselwertes (512)

und wobei das Verfahren ferner umfasst, dass beide Parteien bezüglich der sicheren Kommunikation, d. h. der Prozessor (120) und die externe Vorrichtung, den Masterschlüssel in gleicher Weise ändern.

2. Verfahren (500) von Prozedur c) nach Anspruch 1, wobei das Erzeugen eines neuen Masterschlüsselwertes (512) das Verschlüsseln eines konstanten Wertes (520) unter Verwendung des Masterschlüsselwertes (510) umfasst.

3. Verfahren (200) nach Anspruch 1, wobei in Prozedur a) das Erzeugen des neuen Masterschlüsselwertes (212) das Anwenden einer XOR-Funktion auf den ersten abgeleiteten Sitzungsschlüssel und mindestens den zweiten abgeleiteten Sitzungsschlüssel (242), die unter Verwendung des Masterschlüsselwertes (210) abgeleitet werden, umfasst.

4. Verfahren (200, 300) nach einem vorhergehenden Anspruch, wobei das Ableiten eines Sitzungsschlüssels (240, 242; 340, 342) unter Verwendung des Masterschlüsselwertes (210, 310) umfasst:
Verschlüsseln eines konstanten Wertes (220, 222; 320, 322) unter Verwendung des Masterschlüsselwertes (210, 310).

5. Verfahren (200, 300, 500) nach einem vorhergehenden Anspruch, wobei das Bestimmen eines Masterschlüsselwertes (210, 310, 510) zur Verwendung in sicheren Kommunikationen (360, 560) mit einer externen Vorrichtung umfasst:
Verhandeln eines Masterschlüsselwertes (210, 310, 510) mit der externen Vorrichtung, wobei dies das Verhandeln mit einem sicheren Element (140) umfasst, das Teil der gleichen Vorrichtung ist wie der Prozessor (120).

6. Nichtflüchtiges maschinenlesbares Medium (160), das Anweisungen (168) aufweist, die bei Ausführung auf zwei Computern bewirken, dass die zwei Computer das Verfahren nach einem vorhergehenden Anspruch durchführen.

7. Nichtflüchtiges maschinenlesbares Medium (160) nach Anspruch 6, wobei das nichtflüchtige maschinenlesbare Medium (160) mit Anweisungen zur Ausführung durch ein sicheres Element (140) innerhalb einer Host-Vorrichtung (100) codiert ist, die einer der zwei Computer ist.

8. System zum Durchführen sicherer Kommunikation über einen sicheren Kommunikationskanal zwischen einem Hardwaresystem (100) und einer externen Vorrichtung, wobei das Hardwaresystem (100) einen Prozessor (120) enthält, wobei der Prozessor (120) und die externe Vorrichtung dazu ausgebildet sind
einen Masterschlüsselwert (210, 310, 510) zur Verwendung in sicheren Kommunikationen (260, 360, 560) mit der externen Vorrichtung zu bestimmen,
sowohl in dem Hardwaresystem (100) als auch in der externen Vorrichtung einen ersten Sitzungsschlüssel (240, 340, 540) unter Verwendung des Masterschlüsselwertes (210, 310, 510) periodisch abzuleiten und einen neuen Masterschlüsselwert (212, 312, 512) auf Grundlage des Masterschlüsselwertes (210, 310, 510) periodisch zu erzeugen und den aktuellen Masterschlüsselwert (210, 310, 510) zu löschen,
wobei der Prozessor (120) und die externe Vorrichtung ferner dazu ausgebildet sind, den ersten Sitzungsschlüssel (240, 340, 540) unter Verwendung des Masterschlüsselwertes (210, 310, 510) periodisch abzuleiten und den neuen Masterschlüsselwert (212, 312, 512) auf Grundlage des Masterschlüsselwertes (210, 310, 510) periodisch zu erzeugen, gemäß einer der folgenden Prozeduren a), b) und c):

a) Ableiten (232) eines zweiten Sitzungsschlüssels (242) unter Verwendung des Masterschlüsselwertes (210) und Kombinieren (250) des abgeleiteten ersten Sitzungsschlüssels (240) und des abgeleiteten zweiten Sitzungsschlüssels (242), um dadurch den neuen Masterschlüsselwert (212) zu erzeugen;
b) Ableiten (332) eines zweiten Sitzungsschlüssels (342) unter Verwendung des Masterschlüsselwertes (310), Kombinieren des Masterschlüsselwertes (310) und des abgeleiteten ersten Sitzungsschlüssels (340), um ein kombiniertes Ergebnis zu erbringen, und Kombinieren (352) des kombinierten Ergebnisses und des abgeleiteten zweiten Sitzungsschlüssels (342), um dadurch den neuen Masterschlüsselwert (312) zu erzeugen; und
c) Erzeugen (530) des neuen Masterschlüsselwertes (512) unter Verwendung des Masterschlüsselwertes (510) und Ableiten des ersten oder eines zweiten Sitzungsschlüssels (540) durch Kombinieren (550) des Masterschlüsselwertes (510) und des neuen Masterschlüsselwertes (512)

und wobei beide Parteien bezüglich der sicheren Kommunikation, d. h. der Prozessor (120) und die externe Vorrichtung, dazu ausgebildet sind, den Masterschlüssel in gleicher Weise zu ändern.

9. System von Prozedur c) nach Anspruch 8, wobei beim Erzeugen eines neuen Masterschlüsselwertes (212, 312, 412, 512) der Prozessor (120) dazu ausgebildet ist, einen konstanten Wert (220, 320, 420, 520) unter Verwendung des Masterschlüsselwertes (210, 310, 410, 510) zu verschlüsseln.

10. System nach Anspruch 8, wobei in Prozedur a) das Erzeugen des neuen Masterschlüsselwertes (212) das Anwenden einer XOR-Funktion auf den ersten abgeleiteten Sitzungsschlüssel und mindestens den zweiten abgeleiteten Sitzungsschlüssel (242), die unter Verwendung des Masterschlüsselwertes (210) abgeleitet werden, umfasst.

11. System nach Anspruch 8, wobei beim Ableiten eines Sitzungsschlüssels (240, 242; 340, 342) unter Ver-

wendung des Masterschlüsselwertes (210, 310) der Prozessor (120) dazu ausgebildet ist, einen konstanten Wert (220, 222; 320, 322) unter Verwendung des Masterschlüsselwertes (210, 310) zu verschlüsseln.

Revendications

1. Procédé (200, 300, 500) destiné à être exécuté par un processeur (120) d'un système matériel (100) pour réaliser une communication sécurisée par le biais d'un canal de communication sécurisé avec un dispositif externe, le procédé comprenant :

la détermination d'une valeur de clé maîtresse (210, 310, 510) destinée à être utilisée dans des communications sécurisées (260, 360, 560) avec le dispositif externe ;
à la fois dans le système matériel (100) et dans le dispositif externe, la dérivation périodique d'une première clé de session (240, 340, 540) à l'aide de la valeur de clé maîtresse (210, 310, 510), et la génération périodique d'une nouvelle valeur de clé maîtresse (212, 312, 512) sur la base de la valeur de clé maîtresse (210, 310, 510) ; et
la suppression de la valeur de clé maîtresse (210, 310, 510),
la dérivation périodique de la première clé de session (240, 340, 540) à l'aide de la valeur de clé maîtresse (210, 310, 510) et la génération périodique de la nouvelle valeur de clé maîtresse (212, 312, 512) sur la base de la valeur de clé maîtresse (210, 310, 510) comprenant l'une des procédures a), b) et c) suivantes :

a) la dérivation (232) d'une deuxième clé de session (242) à l'aide de la valeur de clé maîtresse (210), et la combinaison (250) de la première clé de session dérivée (240) et de la deuxième clé de session dérivée (242) pour ainsi générer la nouvelle valeur de clé maîtresse (212) ;
b) la dérivation (332) d'une deuxième clé de session (342) à l'aide de la valeur de clé maîtresse (310), la combinaison de la valeur de clé maîtresse (310) et de la première clé de session dérivée (340) pour produire un résultat combiné, et la combinaison (352) du résultat combiné et de la deuxième clé de session dérivée (342) pour ainsi générer la nouvelle valeur de clé maîtresse (312) ; et
c) la génération (530) de la nouvelle valeur de clé maîtresse (512) à l'aide de la valeur de clé maîtresse (510), et la dérivation de la première ou d'une deuxième clé de session (540) par combinaison (550) de la valeur de clé maîtresse (510) et de la nouvelle valeur de clé maîtresse (512),

et le procédé comprenant en outre la modification, de la même manière, de la clé maîtresse par les deux parties à la communication sécurisée, à savoir le processeur (120) et le dispositif externe.

2. Procédé (500) selon la procédure c) selon la revendication 1, dans lequel la génération d'une nouvelle valeur de clé maîtresse (512) comprend le chiffrement d'une valeur constante (520) à l'aide de la valeur de clé maîtresse (510).

3. Procédé (200) selon la revendication 1, dans lequel, dans la procédure a), la génération de la nouvelle valeur de clé maîtresse (212) comprend l'application d'une fonction OU exclusif à la première clé de session dérivée et au moins à la deuxième clé de session dérivée (242) dérivées à l'aide de la valeur de clé maîtresse (210).

4. Procédé (200, 300) selon l'une quelconque des revendications précédentes, dans lequel la dérivation d'une clé de session (240, 242 ; 340, 342) à l'aide de la valeur de clé maîtresse (210, 310) comprend : le chiffrement d'une valeur constante (220, 222 ; 320, 322) à l'aide de la valeur de clé maîtresse (210, 310) .

5. Procédé (200, 300, 500) selon l'une quelconque des revendications précédentes, dans lequel la détermination d'une valeur de clé maîtresse (210, 310, 510) destinée à être utilisée dans des communications sécurisées (360, 560) avec un dispositif externe comprend :
la négociation d'une valeur de clé maîtresse (210, 310, 510) avec le dispositif externe, comprenant la négociation avec un élément sécurisé (140) faisant partie du même dispositif que le processeur (120).

6. Support non transitoire lisible par machine (160) comprenant des instructions (168) qui, lorsqu'elles sont exécutées sur deux ordinateurs, amènent les deux ordinateurs à réaliser le procédé selon l'une quelconque des revendications précédentes.

7. Support non transitoire lisible par machine (160) selon la revendication 6, lequel support non transitoire lisible par machine (160) est codé avec des instructions destinées à être exécutées par un élément sécurisé (140) au sein d'un dispositif hôte (100) formant l'un des deux ordinateurs.

8. Système destiné à réaliser une communication sécurisée par le biais d'un canal de communication sécurisé entre un système matériel (100) et un dispositif externe, le système matériel (100) comportant un processeur (120), le processeur (120) et le dispositif externe étant configurés

pour déterminer une valeur de clé maîtresse (210, 310, 510) destinée à être utilisée dans des communications sécurisées (260, 360, 560) avec le dispositif externe,

à la fois dans le système matériel (100) et dans le dispositif externe, pour dériver périodiquement une première clé de session (240, 340, 540) à l'aide de la valeur de clé maîtresse (210, 310, 510), et pour générer périodiquement une nouvelle valeur de clé maîtresse (212, 312, 512) sur la base de la valeur de clé maîtresse (210, 310, 510), et

pour supprimer la valeur de clé maîtresse courante (210, 310, 510),

le processeur (120) et le dispositif externe étant configurés en outre pour dériver périodiquement la première clé de session (240, 340, 540) à l'aide de la valeur de clé maîtresse (210, 310, 510) et pour générer périodiquement la nouvelle valeur de clé maîtresse (212, 312, 512) sur la base de la valeur de clé maîtresse (210, 310, 510) selon l'une des procédures a), b) et c) suivantes :

a) la dérivation (232) d'une deuxième clé de session (242) à l'aide de la valeur de clé maîtresse (210), et la combinaison (250) de la première clé de session dérivée (240) et de la deuxième clé de session dérivée (242) pour ainsi générer la nouvelle valeur de clé maîtresse (212) ;

b) la dérivation (332) d'une deuxième clé de session (342) à l'aide de la valeur de clé maîtresse (310), la combinaison de la valeur de clé maîtresse (310) et de la première clé de session dérivée (340) pour produire un résultat combiné, et la combinaison (352) du résultat combiné et de la deuxième clé de session dérivée (342) pour ainsi générer la nouvelle valeur de clé maîtresse (312) ; et

c) la génération (530) de la nouvelle valeur de clé maîtresse (512) à l'aide de la valeur de clé maîtresse (510), et la dérivation de la première ou d'une deuxième clé de session (540) par combinaison (550) de la valeur de clé maîtresse (510) et de la nouvelle valeur de clé maîtresse (512),

et les deux parties à la communication sécurisée, à savoir le processeur (120) et le dispositif externe, étant configurées pour modifier, de la même manière, la clé maîtresse.

9. Système selon la procédure c) selon la revendication 8, dans lequel, lors de la génération d'une nouvelle valeur de clé maîtresse (212, 312, 412, 512), le processeur (120) est configuré pour chiffrer une valeur constante (220, 320, 420, 520) à l'aide de la valeur de clé maîtresse (210, 310, 410, 510).

10. Système selon la revendication 8, dans lequel, dans la procédure a), la génération de la nouvelle valeur de clé maîtresse (212) comprend l'application d'une fonction OU exclusif à la première clé de session dérivée et au moins à la deuxième clé de session dérivée (242) dérivées à l'aide de la valeur de clé maîtresse (210).

11. Système selon la revendication 8, dans lequel, lors de la dérivation d'une clé de session (240, 242 ; 340, 342) à l'aide de la valeur de clé maîtresse (210, 310), le processeur (120) est configuré pour chiffrer une valeur constante (220, 222 ; 320, 322) à l'aide de la valeur de clé maîtresse (210, 310).

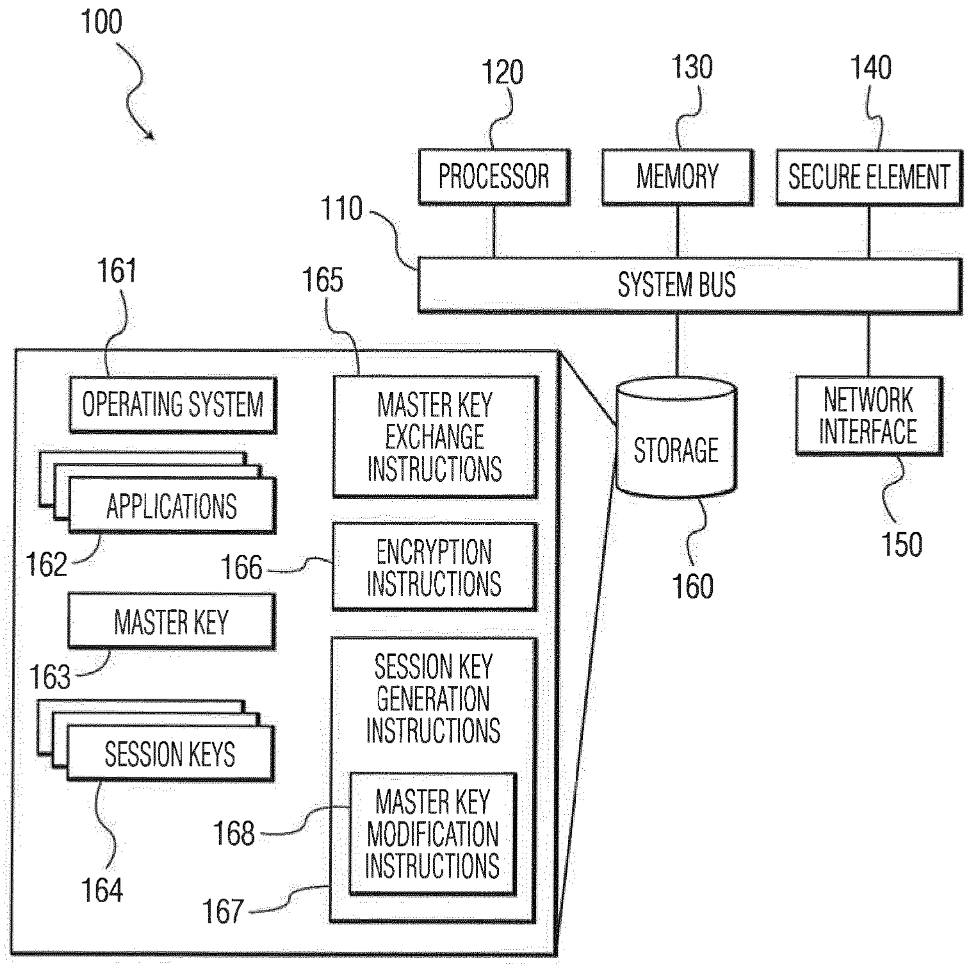
100
120
130
140
PROCESSOR
MEMORY
SECURE ELEMENT
110
SYSTEM BUS
161
165
OPERATING SYSTEM
MASTER KEY EXCHANGE INSTRUCTIONS
STORAGE
NETWORK INTERFACE
APPLICATIONS
162
166
ENCRYPTION INSTRUCTIONS
150
160
MASTER KEY
163
SESSION KEY GENERATION INSTRUCTIONS
SESSION KEYS
168
MASTER KEY MODIFICATION INSTRUCTIONS
164
167

FIG. 1

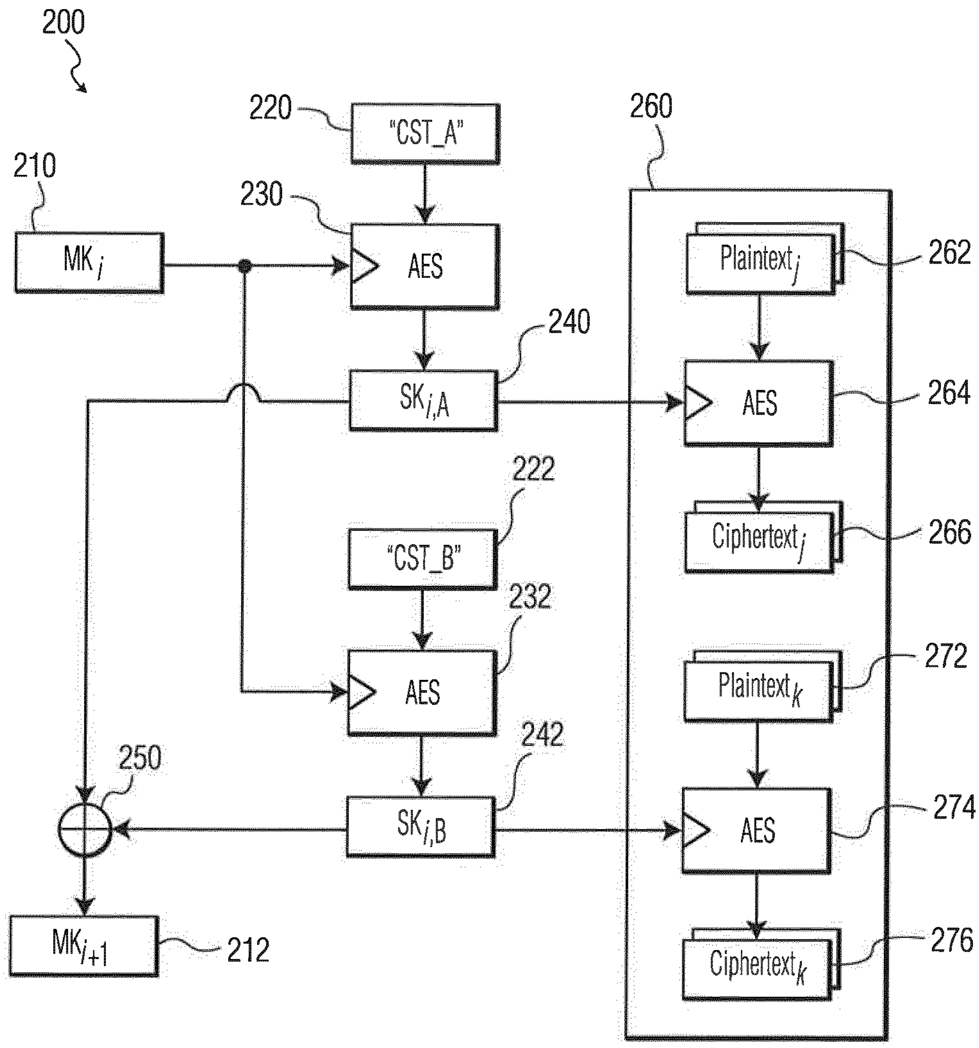
200
210
MK$_i$
220
"CST_A"
230
AES
240
SK$_{i,A}$
222
"CST_B"
232
AES
242
SK$_{i,B}$
250
MK$_{i+1}$
212
260
Plaintext$_j$
262
AES
264
Ciphertext$_j$
266
Plaintext$_k$
272
AES
274
Ciphertext$_k$
276

FIG. 2

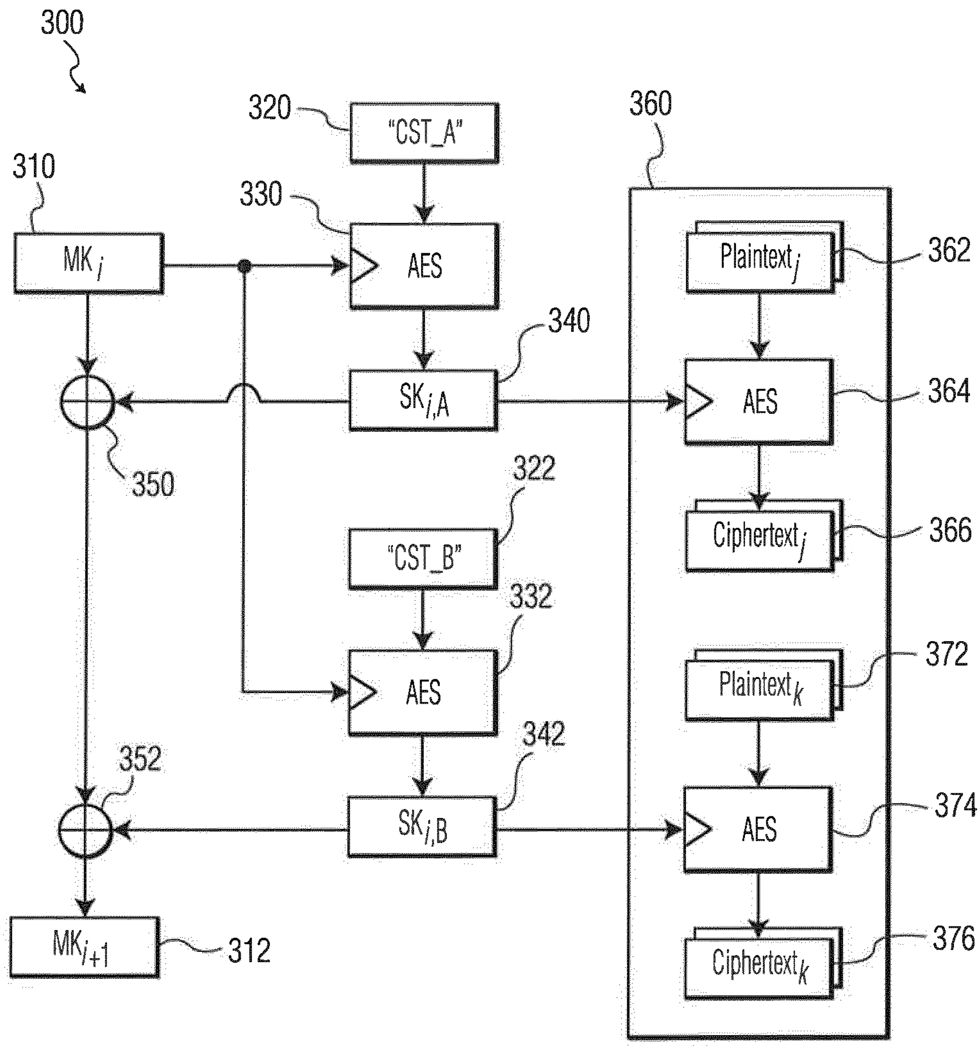
300
320
"CST_A"
360
310
330
MK $_i$
AES
Plaintext $_j$
362
340
SK $_{i,A}$
AES
364
350
322
Ciphertext $_j$
366
"CST_B"
332
AES
Plaintext $_k$
372
342
352
SK $_{i,B}$
AES
374
MK $_{i+1}$
312
Ciphertext $_k$
376

FIG. 3

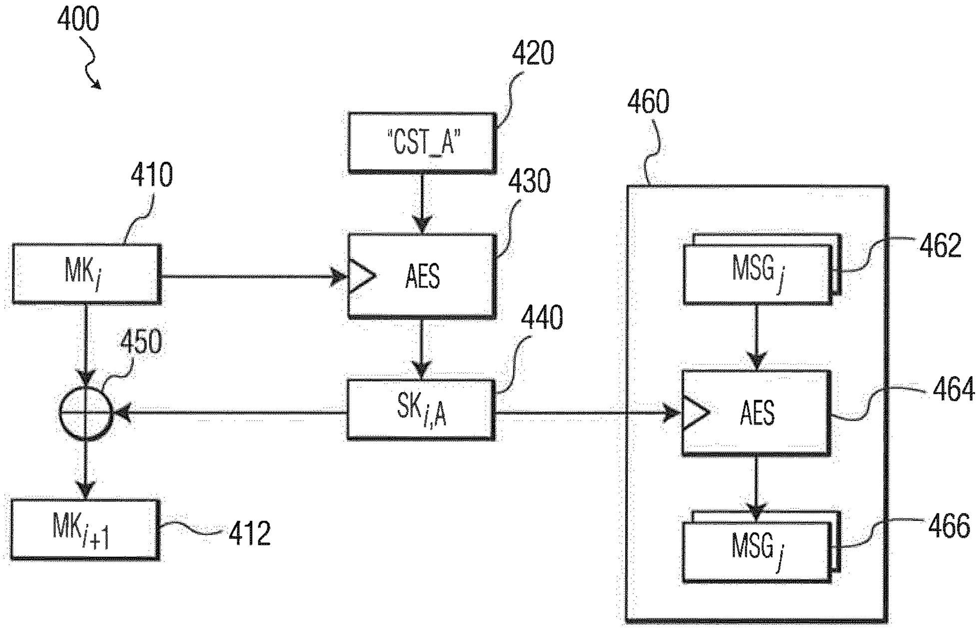
400
420
"CST_A"
460
410
430
MK$_i$
AES
462
MSG$_j$
450
440
SK$_{i,A}$
464
AES
MK$_{i+1}$
412
466
MSG$_j$

FIG. 4

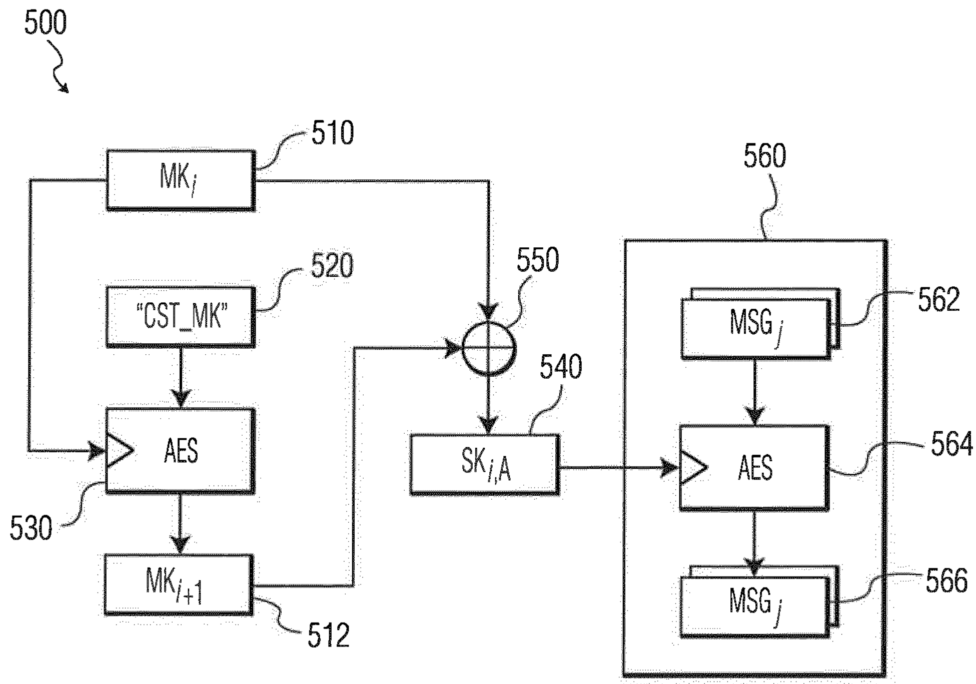
500
510
MK$_i$
560
520
550
"CST_MK"
562
MSG$_j$
540
AES
SK$_{i,A}$
564
AES
530
MK$_{i+1}$
512
566
MSG$_j$

FIG. 5

REFERENCES CITED IN THE DESCRIPTION

This list of references cited by the applicant is for the reader's convenience only. It does not form part of the European patent document. Even though great care has been taken in compiling the references, errors or omissions cannot be excluded and the EPO disclaims all liability in this regard.

Patent documents cited in the description

- US 20070003063 A1 **[0003]**